# Supporting the Curation of Twitter User Lists

**Derek Greene, Fergal Reid**
School of Computer Science & Informatics,
University College Dublin, Ireland
derek.greene@ucd.ie, fergal.reid@gmail.com

**Gavin Sheridan**
Storyful
Dublin, Ireland
gavin.sheridan@storyful.com

**Pádraig Cunningham**
School of Computer Science & Informatics,
University College Dublin, Ireland
padraig.cunningham@ucd.ie

## Abstract

Twitter introduced *lists* in late 2009 as a means of curating tweets into meaningful themes. Lists were quickly adopted by media companies as a means of organising content around news stories. Thus the curation of these lists is important, they should contain the key information gatekeepers and present a balanced perspective on the story. Identifying members to add to a list on an emerging topic is a delicate process. From a network analysis perspective there are a number of *views* on the Twitter network that can be explored, *e.g.* followers, retweets mentions etc. We present a process for integrating these views in order to recommend authoritative commentators to include on a list. This process is evaluated on manually curated lists about unrest in Bahrain and the Iowa caucuses for the 2012 US election.

## 1 Introduction

Media outlets that leverage the content produced by users of social media sites can now break or cover stories as they evolve on the ground, in real time (e.g. videos, photographs, tweets). However, a signicant issue arises when trying to (a) identify content around a breaking news story in a timely manner (b) monitor the proliferation of content on a certain news event over a period of time, and (c) ensure that content is reliable and accurate. Storyful[1] is a social media news agency established in 2009 with the aim of filtering news, or newsworthy content, from the vast quantities of noisy data that streams through social networks. To this end, Storyful invests significant time into the manual curation of content on social media networks, such as Twitter and YouTube. In some cases this involves identifying "gatekeepers" who are prolific in their ability to locate, filter and monitor news from eyewitnesses.

Twitter users can organise the users they follow into Twitter *lists*. Storyful maintains lists of users relevant to a given news story, as a means of monitoring breaking news related to that story. Often these stories generate community-decided hashtags (*e.g. #occupywallstreet*) – but even with small news events, using such hashtags to track the evolution of a story becomes difficult. Spambots quickly intervene, while users with no proximity (in space, time or expertise) to the news story itself drown out other voices. Manual curation of lists is one way to overcome this problem, but is time consuming, and risks incomplete coverage. In order to support the list curation process, we propose methods for identifying the important users that form the "community" around a news story on Twitter. Specifically, given a small seed list of users supplied by a domain expert, we are interested in using network analysis techniques to expand this set to produce a user list that provides

---

[1] http://www.storyful.com

comprehensive coverage of the story. The motivation is that the members of this list will provide additional valuable content relating to the story.

A number of authors have considered the related problem of producing personal recommendations for additional users to follow on Twitter, either by following user links or performing textual analysis of tweet content. Hannon *et al.* [3] proposed a set of techniques for producing personal recommendations on users to follow, based on the similarity of the aggregated tweets or "profiles" of users that are connected to the ego in the Twitter social graph. Such techniques have primarily relied on a single view of the network to produce suggestions. However, we can view the same Twitter network from a range of different perspectives. For instance, Conover *et al.* [2] performed an analysis of Twitter data based on references to other Twitter screen names in a tweet, while researchers have also looked at the diffusion of content via *retweets* to uncover the spread of memes and opinions on Twitter [2, 6]. The idea is that both *mentions* and *retweets* provide us with some insight of the differing interactions between microblogging users.

In Section 2 we describe a set of recommendation criteria **and** network exploration methods used to support user list curation on Twitter. Rather than using a single view of the network to produce recommendations, we employ a *multi-view* approach that produces user rankings based on different graph representations of the Twitter network surrounding a given user list, and combines them using an SVD-based aggregation approach [10]. Information from multiple views is also used to control the exploration of the Twitter network – this is an important consideration due to the limitations surrounding Twitter data access. To verify the accuracy of the resulting recommendations, in Section 3 we describe experiments performed on a previously-curated Twitter list relating to coverage of the Iowa caucuses in advance of the 2012 US Presidential Election. In Section 4 we investigate whether a "silo" effect arises in cases where a user list is expanded from an initial seed list with a strong bias towards a particular perspective on a story. We do this by evaluating the proposed recommendation techniques on subsets of a previously-curated list covering the current political situation in Bahrain. This study motivates further work in this area, which is discussed in Section 5.

## 2 Methods

### 2.1 Bootstrapping

We now describe our proposed system for supporting user list curation. The initial input to the system is a seed list of one or more users that have been manually labelled as being relevant to a particular news story. Once a seed list has been supplied, the first operation of the system involves a *bootstrapping phase*, which retrieves follower ego networks around all seed list members. Other information regarding these users is also retrieved – such as user list membership information and a limited number of tweets. The extent of the exploration process can be controlled by setting an upper limit for the number of links to follow and tweets to retrieve – these parameters control the trade-off between network exploration depth and the number of queries required. The latter is an important consideration, not only in terms of running time, but also due to the fact that Twitter employs a quota system that limits the number of permitted API queries that can be made per hour.

After the bootstrap phase, the system will have two disjoint lists for the news story. The *core set* contains curated Twitter accounts, initially this corresponds to the members of the seed list. The *candidate set* contains Twitter accounts that are not in the core set, but exist in the wider network around the core – some of these users may potentially be relevant for curation, while others will be spurious. Initially this will consist of the new non-seed users that were found during the bootstrap phase.

### 2.2 Recommending Users

In the subsequent *recommendation phase*, a ranked list of the $r$ top users from the candidate set is produced. Firstly, we produce individual rankings using a number of criteria applied to different graphs, each representing a different view of the same network. The motivation is that each view potentially captures a different aspect of the relations between Twitter users around a given news story. We construct four different views:

1. *Core friend graph*: This is a directed graph which contains nodes representing all users in the core set, along with the non-core users who they follow.

2. *Core mention graph:* As an alternative network view, we analyse the non-core users mentioned by the users in the core set. Specifically, an edge links from a core node A to a non-core node B if A has mentioned B in at least one tweet – the weight of the edge corresponds to the number of tweets. The idea here is that this directed, weighted mention graph is a proxy for the dialogue between these Twitter users.

3. *Core retweet graph:* We also analyse retweeting activity by core users involving tweets originally posted by non-core users. This involves the construction of a weighted, directed graph, where an edge links from core node A to non-core node B if A has retweeted B's tweets at least once – the weight of the edge corresponds to the number of retweets.

4. *Weighted co-listed graph:* Another alternative view, which has not been widely explored in the literature, is to look at relations based on the aggregation of co-assignments to Twitter user lists. At an aggregate level, this could be regarded as a form of crowd-sourced curation, where the assumption is that related pairs of users will be more frequently assigned to the same list than users who have dissimilar to one another. Based on this idea, we construct a weighted, undirected graph as follows. For each user list that has been identified, we measure the overlap $w$ between the list's members and the core set using the Jaccard set similarity measure [4]. If $w > 0$ then, for each unique core/non-core pair of users in the user list, we create an edge between these two users with weight $w$. If an edge between the users already exists, we increment the weight on the edge by $w$.

The criteria that we use on these graphs are as follows:

1. *In-degree*: A simple approach for directed graphs is to look at the in-degree centrality of each Twitter user. For weighted graphs, we calculate the sum of the weights on incoming edges.

2. *Normalised in-degree:* Using standard in-degree centrality can potentially lead to the selection of high-degree Twitter users who are not specialised in a particular geographic or topical area. Our solution has been to introduce a normalisation factor to reduce the impact of high degree nodes. The normalisation approach is similar to standard log-based TF-IDF term weighting functions that are widely applied in text mining to reduce the influence of frequently-occurring terms [7]. The normalised follower count value for the user $u$ is defined as:

$$\text{nfc}(u) = \log\left(seed\_followers(u)\right) \cdot \log\left(\frac{max\_followers}{all\_followers(u)}\right) \quad (1)$$

   where

   - $seed\_followers(u)$ = the number of users in the core set that follow the user $u$.
   - $all\_followers(u)$ = the total number of all users following the user $u$ on Twitter.
   - $max\_followers$ = a scaling factor, defined to be the largest number of Twitter followers among any of the core and non-core users.

3. *HITS with priors*: The HITS algorithm, originally proposed by [5], has been widely used to assign hub and authority scores to each mode in graph, depending upon its the topology. We can use the authority scores applied to a Twitter network to identify key users in that network. Since we wish to focus on authority relative to our pre-curated core list, we use the variation of HITS proposed by [9], which introduces prior probabilities for each node. Specifically, each of the $m$ users in the core list is given an initial probability $1/m$, while the other non-core nodes are given an initial probability of 0.

Naturally, certain criteria are only meaningful when applied to certain graphs. For the purpose of the evaluations described in this paper, we use the following five combinations:

- Normalised degree applied to the core friend graph.
- HITS with priors applied to the core friend graph.
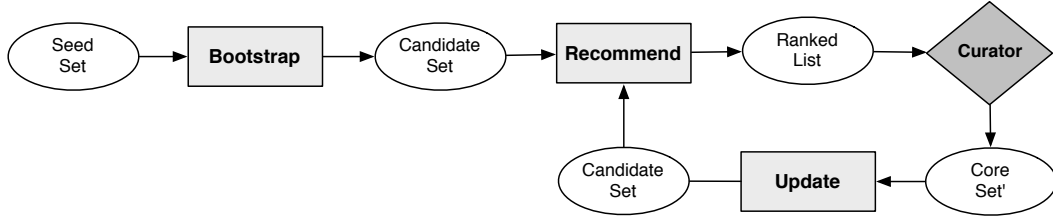- Weighted in-degree applied to the co-listed, mention, and retweet graphs.

3

Figure 1: Overview of curation support system, illustrating the workflow between the *bootstrap*, *recommendation*, and *update* phases.

### 2.2.1 Combining Rankings

The various graph/criterion combinations can potentially produce rankings of users that differ significantly. To combine rankings, we use SVD-based aggregation, which has previously been shown to be effective for this task [10]. We construct a matrix $\mathbf{X}$ from the ranks (rather than the raw scores), with users on the rows and rankings on the columns. We then apply SVD to this matrix and extract the first left singular vector. The values in this vector provide aggregated scores for the users. By arranging these values in descending order, we can produce a final ranking of users. We select the top $r$ users to form our list of user recommendations. Finally, we can also apply additional filtering of recommendations based on a minimum tweet count filter and a filter to remove users who have not tweeted within a given time period.

After a set of recommendations has been generated, the ranked list of suggested users would be presented to a human curator, who could then select a subset to migrate to the core set (*i.e.* to augment the existing Twitter user list). The use of a "human in the loop" in the proposed system resembles the role of the oracle in active learning algorithms for classification [8].

### 2.3 Network Exploration

Once the core set has been modified, the system enters the *update phase*, which modifies the current copy of the network to reflect (a) changes in membership of the core set, and (b) any changes in the Twitter network since the last update (*e.g.* addition/removal of follower links, new tweets). Specifically, the network is explored using a process based both on the follower graph and also on tweet content:

- For the current core set, retrieve their friend/follower links, user list memberships, and recent tweets for all set members (*i.e.* same process as in the bootstrap phase).

- For the last set of recommended users who were not migrated to the core set, retrieve their friend/follower links, user list memberships, and recent tweets.

- For the set of $m$ users who were most frequently mentioned in tweets posted by the core set, retrieve their friend/follower links, user list memberships, and recent tweets.

Again the extent of the exploration for each of the above can be controlled by setting maximum values for the number of links to follow and tweets to retrieve. Once the local copy of the network has been update, the data then feeds back to the recommendation phase and another iteration of the recommendation-selection-update process is executed. A visual overview of the complete curation system process is shown in Fig. 1.

## 3 Case Study 1: Iowa

### 3.1 Experimental Setup

First, we evaluated the proposed recommendation system on a Twitter user list previously curated by Storyful, covering Iowa politics during the 2012 US Presidential Primaries[2]. At the time of

---

[2]`http://twitter.com/#!/trailmix12/iowa`

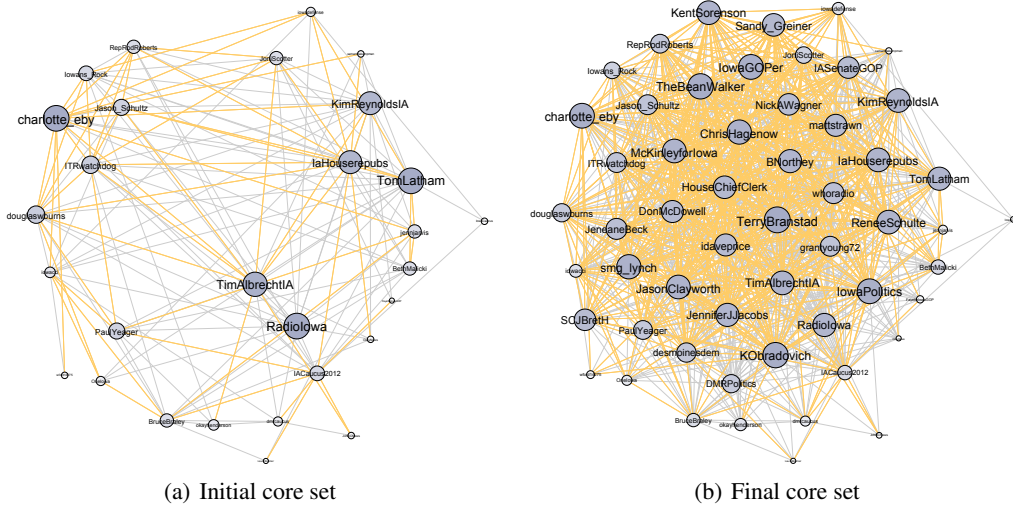|                        |                        |
| :--------------------: | :--------------------: |
| (a) Initial core set   | (b) Final core set     |

Figure 2: Induced subgraph of the follower graph for the core set members in the Iowa dataset #1 after (a) the initial bootstrap phase, (b) six complete iterations. Larger nodes with a more saturated colour are indicative of nodes with a higher in-degree (*i.e.* users with more followers within the core set). Highlighted edges indicate reciprocated follower links between users. Layout positions are preserved for both figures.

initial data collection – 16 September 2011 – this list contained 128 unique users. To evaluate the robustness of the user recommendation process, we use cross validation, randomly dividing the complete Iowa user list into four disjoint datasets, each containing 32 users. As an example, the subgraph induced by the core set on the follower graph of Iowa dataset #1 is shown in Fig. 2(a) – the positions of nodes were calculated using the force directed layout implementation provided by Gephi [1].

In our experiments we applied the workflow shown in Fig. 1 to each of the sets individually for six *recommendation-selection-update* iterations after the initial bootstrapping phase. Note that no information was shared between the runs. The extent of network exploration during the update phase was controlled using the following constraints:

- A maximum of up to 1,000 friend/follower links were retrieved per user at a given iteration.
- A maximum of up to 1,000 user lists were retrieved per user at a given iteration.
- A maximum of up to 1,000 tweets was retrieved per user at a given iteration.
- Very high-degree users $> 50,000$ friends and/or 50,000 followers were filtered.

To generate recommendations, we used the views and criteria as described in Section 2. We filtered the recommendations to remove those users who had not tweeted in the previous two weeks and/or those who had posted fewer than 25 tweets in total. At each iteration we generated $r = 50$ recommendations – by the final iteration, users were selected from a complete candidate set with average size of $\approx 62k$ users. At this stage, we had also collected an average $\approx 63k$ tweets and $\approx 138k$ follower links for each dataset. In place of a manual curator, after each complete iteration we automatically selected the top five highest ranked users (based on SVD aggregation) to add to the core set. The six iterations thus yielded 30 additional core users for each of the four sets. As an example, the final expanded core set for Iowa dataset #1 is shown in Fig. 2(b). It is interesting to observe that several high-degree nodes were added to the core set, such as the user @*TerryBranstad*, the official account of the Governor of Iowa.

## 3.2 Discussion

Next, to quantitatively validate the relevance of the recommendations produced by our proposed techniques, we use two measures that are frequently used in information retrieval tasks: *precision*

| Iteration | Precision | | | | | Recall | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Set 1 | Set 2 | Set 3 | Set 4 | Mean | Set 1 | Set 2 | Set 3 | Set 4 | Mean |
| 1 | 0.95 | 0.97 | 0.97 | 1.00 | **0.97** | 0.27 | 0.28 | 0.28 | 0.29 | **0.28** |
| 2 | 0.93 | 0.98 | 0.95 | 1.00 | **0.96** | 0.30 | 0.32 | 0.31 | 0.33 | **0.32** |
| 3 | 0.94 | 0.96 | 0.96 | 0.98 | **0.96** | 0.34 | 0.35 | 0.35 | 0.36 | **0.35** |
| 4 | 0.90 | 0.94 | 0.96 | 0.96 | **0.94** | 0.37 | 0.38 | 0.39 | 0.39 | **0.38** |
| 5 | 0.88 | 0.93 | 0.93 | 0.95 | **0.92** | 0.39 | 0.41 | 0.41 | 0.42 | **0.41** |
| 6 | 0.82 | 0.92 | 0.89 | 0.89 | **0.88** | 0.40 | 0.45 | 0.43 | 0.43 | **0.43** |

Table 1: Precision and recall scores for four randomly-selected subsets of the Storyful Iowa user list, for six complete *recommendation-selection-update* iterations.

and *recall*. The results for the four datasets across all six iterations are listed in Table 1. We observe that, in terms of recall, increasing the user list size by 30 accounts does not lead to a significant fall in precision – average precision relative to the complete original list remains at 0.88 by iteration six. Meanwhile, recall increases steadily in all cases – the average is 0.43. Note the maximum achievable recall by iteration six is 0.48 (*i.e.* 62 out of the 128 users are returned), and is lower in previous iterations.

In general, we observe that the Iowa user list studied here consists of a relatively homogeneous group of users pertaining to a story with a relatively narrow focus – the users are predominantly Republicans involved in the Iowa caucuses. Therefore, unlike the study in [2] which analysed Twitter relations across the entire country during 2010 US midterm elections, here a pronounced partisan divide is not evident.

# 4 Case Study 2: Bahrain

## 4.1 Experimental Setup

For our second study, we analyse a dataset with significantly different characteristics. As a seed list we use a Twitter list covering the current political situation in Bahrain which was also manually curated by Storyful[3]. As of 27 September 2011, this list contained 51 users. A small number of these have a "loyalist" or "pro-government" stance, while the remaining users could be regarded as being either "non-loyalist", or "neutral" observers with an interest in Bahrain. This natural division in the seed list raises an interesting question – does starting with a seed list that takes a particular stance on a given news story lead to the construction of localised network "silos", which may lead an automated system to give biased user recommendations?

To investigate this, we generate recommendations based on a seed list *Bahrain-L* containing a subset of 14 users that have been putatively labelled as "loyalist". We ran four complete iterations using the same exploration constraints, filters, and selection mechanism used in the previous evaluation. This resulted in a core set containing 34 users, a candidate set of 51,114 users, 138,777 follower links, and 53,450 unique tweets.

## 4.2 Discussion

Fig. 3(a) shows the subgraph induced by the original complete curated list of 51 users on the follower graph – the split between the "loyalist" users and the other users is evident from the positions calculated by force directed layout. In particular, the latter group of users form a densely connected core, while most of the "loyalist" nodes are not well-connected with the rest of the subgraph. Fig. 3(b) shows a subgraph induced by the union of the curated list, with the set of nodes selected based on the recommendation process using *Bahrain-L* alone as the seed set. We observe that none of the 37 "non-loyalist" nodes from the curated list were selected during the four iterations. In contrast, we see that the new users are closely connected with the other "loyalist" users, forming a second dense core. While we might expect this if recommendations were only generated based on follower links, recall that rankings based on mentions and retweets are also being aggregated to select new users.

---

[3] http://twitter.com/#!/storyfulpro/bahrain

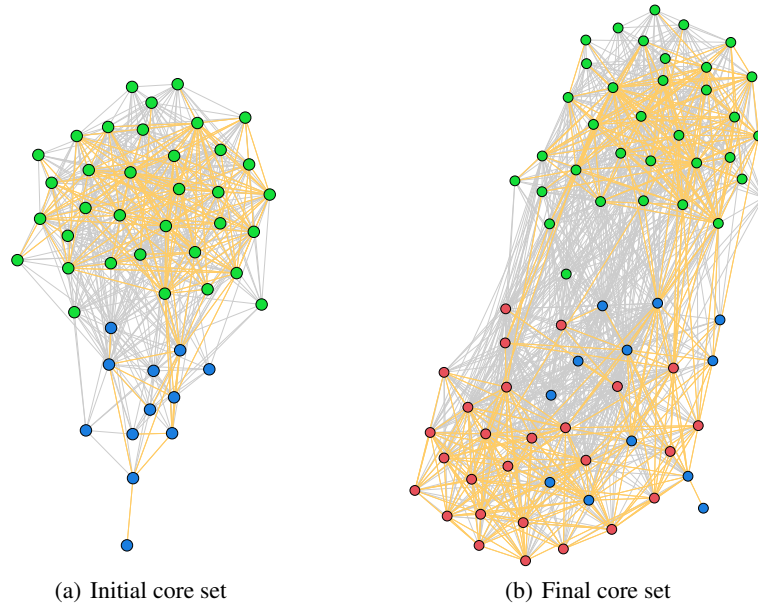|          |          |
|:--------:|:--------:|
| (a) Initial core set | (b) Final core set |

Figure 3: Follower graph for core set members in the Bahrain dataset after (a) the initial bootstrap phase, (b) four complete iterations. Blue nodes denote users in the original user list that are putatively labelled as "loyalist", while the remaining members of the user list are coloured green. The additional nodes that have been selected, based on recommendations using *Bahrain-L* as a seed list, are coloured red.

In fact, the addition of these rankings appears to further compound the "silo" effect which is evident from Fig. 4.2.

Our analysis suggests that there is little interaction on Twitter between users with differing stances on the political situation in Bahrain. On the one hand, this highlights weakness of the proposed recommendation techniques in the case of stories that are highly-polarised. Alternative criteria, which emphasise diversity over homogeneity, may provide a solution – this is analogous to the attempts in active learning to identify diverse examples in order to widely cover the sample space [8]. On the other hand, these results also highlight the continued importance of the role of the curator in (a) selecting a suitably diverse seed list as a starting point, (b) actioning recommendations produced by the system.

## 5 Conclusions

In this paper we have proposed a comprehensive approach for automating aspects of the Twitter list curation process, based on novel network exploration and multi-view recommendation techniques. In the evaluation in Section 3, we showed that, using different starting subsets of a manually-curated list, we can recall the original human annotations while maintaining high precision.

Based on the observations made in Section 4, we suggest that the next major phase of our work will involve exploring the diffusion patterns of newsworthy multimedia resources (*e.g.* links to images and videos) in the network surrounding a user list. For instance, identifying users who are frequently early in *retweet chains* for such resources may help diversify user list recommendations in situations where the "silo" effect is pronounced, such as in the Bahrain case study. In future we also plan to apply the proposed recommendation and network exploration techniques beyond Twitter, looking at multiple views across several different social networks. A key issue here will be the generation of a reliable mapping between users on different networks.

# References

[1] M. Bastian, S. Heymann, and M. Jacomy. Gephi: An open source software for exploring and manipulating networks. In *Proc. International AAAI Conference on Weblogs and Social Media (ICWSM'09)*, pages 361–362, 2009.

[2] M. D. Conover, J. Ratkiewicz, M. Francisco, B. Gonçalves, A. Flammini, and F. Menczer. Political polarization on twitter. In *Proc. 5th Intl. Conference on Weblogs and Social Media (ICWSM'11)*, 2011.

[3] J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proc. 4th ACM conference on Recommender systems*, pages 199–206. ACM, 2010.

[4] P. Jaccard. The distribution of flora in the alpine zone. *New Phytologist*, 11(2):37–50, 1912.

[5] J.M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5):604–632, 1999.

[6] J. Ratkiewicz, M. Conover, M. Meiss, B. Gonçalves, A. Flammini, and F. Menczer. Detecting and tracking political abuse in social media. In *Proc. 5th International AAAI Conference on Weblogs and Social Media (ICWSM'11)*, 2011.

[7] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. Technical Report 87-881, Department of Computer Science, Cornell University, Ithaca, NY, USA, 1987.

[8] B. Settles. Active Learning Literature Survey. Technical Report 1648, University of Wisconsin–Madison, 2009.

[9] S. White and P. Smyth. Algorithms for estimating relative importance in networks. In *Proc. 9th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 266–275. ACM, 2003.

[10] G. Wu, D. Greene, and P. Cunningham. Merging multiple criteria to identify suspicious reviews. In *Proc. 4th ACM Conference on Recommender Systems (RecSys'10)*, 2010.